



Universidade Tecnológica Federal do Paraná

Campus Campo Mourão

Departamento de Computação - DACOM

Prof. Dr. Diego Bertolini

Disciplina: Inteligência Computacional



Projeto 1 - Momentos de HU

- O trabalho deve ser entregue no dia 20 de setembro ;
- Pode ser feito em duplas ;
- Entregar código + relatório ;

Detalhes do Trabalho:

Demonstrar que o Template Matching usando a própria imagem pode ser eficaz em algumas tarefas e ineficiente na maioria delas.

Etapa A:

1. Baixe a base “four shapes” no kaggle: <https://www.kaggle.com/datasets/smeschke/four-shapes>
2. Selecione 10 imagens aleatórias de cada uma das 4 classes ;
3. Utilizando os Momentos Invariantes de HU demonstre que as imagens são mais similares entre a mesma classe e diferentes interclasses ;
 - a. Escolha uma imagem como “*template*” e três imagens da mesma classe como entradas ;
 - b. Calcule os 7 momentos de HU ([código de exemplo](#)) de cada imagem ;
 - c. Utilizando a distância euclidiana, calcule a distância entre o template e a imagem de entrada.

*** No Matlab existe essas funções prontas.

Etapa B:

1. Refaça os experimentos da Etapa A usando a base de folhas ([download](#)) ;
2. Selecione 3 imagens aleatórias de cada uma das 3 classes ;
3. Utilizando os Momentos Invariantes de HU demonstre que as imagens são mais similares entre a mesma classe e diferentes interclasses ;
 - a. Escolha uma imagem como “*template*” e três imagens da mesma classe como entradas ;
 - b. Calcule os 7 momentos de HU (através do código disponível) de cada imagem ;
 - c. Utilizando a distância euclidiana, calcule a distância entre o template e a imagem de entrada.

Avaliação:

- Descreva as taxas de acerto em no máximo 2 páginas. Descreva as vantagens e desvantagens de usar o Template Matching.
 - Modelo .doc ou .tex para escrita do artigo: [link](#)
- Receberão NOTA 0 (ZERO) trabalhos plagiados de qualquer fonte, e/ou com códigos idênticos ou similares.
- Trabalhos que não apresentarem os requisitos mínimos descritos neste documento serão penalizados.

Exemplo de código Octave/Matlab

```
#####
```

```
imagefiles = dir('*.png'); % veja o tipo das imagens e altere, neste caso temos o .png
```

```
[num, z] = size(imagefiles) ; % num vai ter o total de imagens...
```

```
for i = 1 : num
```

```
    image = imread( imagefiles(i).name) ; % carrega a imagem...
```

```
end
```